

Electronic Communications of the EASST
Volume 48 (2011)



Proceedings of the
Fifth International Workshop on
Foundations and Techniques for
Open source Software Certification
(OpennCert 2011)

Quality, Success, Communication and Contribution in Open Source
Software

Sara Fernandes

10 pages

Guest Editors: Luis Soares Barbosa, Dimitrios Settas
Managing Editors: Tiziana Margaria, Julia Padberg, Gabriele Taentzer
ECEASST Home Page: <http://www.easst.org/eceasst/>

ISSN 1863-2122

Quality, Success, Communication and Contribution in Open Source Software

Sara Fernandes¹

¹sarasantos.fernandes@gmail.com

HASLab / INESC TEC , University of Minho

Abstract: Free and open source software projects are often perceived to be of high quality. To a great extent the success of open source software seems to be due to an implicit but effective connection between communication and contributions in its development process. In this paper, we present a snapshot of the state the art on quality and success of Open Source Software (OSS) based on a review of the literature. For each of these concepts, we describe various measures considered in the literature and a number of methods by which they are obtained. Contributions to an Open Source Software (OSS) project are made through communication among developers and users. We elaborate on the concrete notions of communication and contribution in Open Source Software (OSS) and their links.

Keywords: Open Source Software, Quality, Success, Communication, Contribution

1 Introduction

To study the properties of Open Source Software (OSS), we need to consider this phenomenon as a process, both technical and social, not just its end product. The development of open source software is inseparable from the communities that engage in its development. This suggests that a suitable model to study the properties of OSS may be based on the models for Information Systems (IS). Silver et al. defined IS as follows: “Information systems are implemented within an organization for the purpose of improving the effectiveness and efficiency of that organization. Capabilities of the information system and characteristics of the organization, its work systems, its people, and its development and implementation methodologies together determine the extent to which that purpose is achieved”.

Open source software is very often developed in a public, collaborative manner. The qualifier “collaborative” designates a style of software development whose focus is on public availability and communication, mainly via the internet. The collaborative aspect of OSS development is similar to the one used on an IS.

An IS is any combination of information technology and people’s activities using that technology to support operations, management, and decision making. This combination is collaborative and somehow close to the one used on OSS development.

Regarding OSS development as an IS allows us to apply the existing models of information systems success (DeLone and McLean Model) to predict potential success measures for OSS projects. However, OSS projects and IS also differ in significant ways. This makes a number of the predictive measures for the success of IS unsuitable for OSS, while others are difficult to apply in OSS environments.



Many of the predictive measures for the success of IS are based on a vision of system development in an organization that is at the core of an IS. This however does not take into account the unique characteristics of the OSS development environment, embodied in OSS communities. Free open source projects are carried out by volunteers that are members of a OSS community. Due to the free commitment to contribute to the development of any OSS product, the contributions of the volunteers aren't always reliable since volunteers may stop carrying out their duties for a long period of time or, in some cases, never return to them. Also the distributed characteristic of the OSS project makes it impossible to gather all the volunteers at the same time[Mic04].

In this paper, we propose to apply a version of DeLone's model, modified to accommodate what distinguishes OSS projects from IS, to predict the success of OSS projects.

Despite of the similarities and differences between OSS projects and IS, OSS projects can be seen as organizations. It is necessary to think of an OSS community as an organization, to make it possible to apply the measures of success of IS in the model of DeLone and McLean to OSS projects, taking into consideration the fact that there can be more categories/dimensions than the ones presented in the original model. OSS can be seen as an organization but it emerges organically bottom up, as opposed to the top down created-by-design structures of traditional organization where information systems have been studied. Because they are organic and self-organizing, and rely on motivated volunteers, some measures and factors that are relevant for their categorization are different than those for the case of traditional IS.

Open Source Software (OSS) projects are often thought of having the unique characteristic of being developed by volunteers and are perceived to have higher quality. Much attention has been paid to the implications of development by motivated volunteers. In this paper we clarify the concepts of quality, success, contribution, and identify a number of characteristics of communication in OSS projects. We also identify measures that can be applied to assess the quality and success of an OSS product and the distribution of contribution and communication on the development of high quality and successful OSS projects and products. Our goal is to consider different measures of these four properties and to establish relationships between them.

The remainder of this paper is organized as follow. In Section 1 we present the research context. In Section 2, we briefly review the literature on relevant measures for software quality and success in OSS products. Section 3, reviews the literature on communication and contribution in OSS projects, and describes the properties used in the literature to measure communication and contribution. In Section 4, we discuss possible links between the concepts presented in Sections 2 and 3, resorting to the DeLone and McLean Model of IS Success. Based only on the literature, this comparison suggests additional measures that may be incorporated in the development of successful and quality OSS. We conclude in Section 5, by suggesting directions for future research.

2 Research Context

Open Source Software (OSS) plays an increasingly important role in our society. Consequently, it is necessary for OSS to have and maintain a suitable level of quality to justify third-party confidence on its products. Both of these factors (quality and success) are influenced by the fact that OSS allows users and developers to have access to the source code that is developed under

some sort of "open source" license. There are different forms of licenses[UPO11] for OSS that grant substantially different rights to their users. However, we will not discuss the OSS licenses in this paper. In order to have OSS (and/or Free Software)[Per04] that is not only successful but also has quality and impact in the society, it is necessary to understand not only how success and quality can be measured, but also the physical distribution of contributors/developers, the distribution of the contributions made by the developers and users, and certain characteristics of communication that takes place in the OSS development process.

One of the most quoted papers on the phenomena of OSS, is "The Cathedral and the Bazaar" [Ray99] by Eric Raymond. The paper claims that the OSS process is marked by volunteerism, collaboration, and a lack of project structure. It recalls that the development progress depends on the developers free time and because of it, there is no release schedule. Often, there is only an outline of features. And in the limit, the end user ensures quality. Obviously, many OSS projects involve not only volunteers; big companies put their paid employees to work on OSS as well. Companies such as IBM and Sun Microsystems have entire divisions devoted solely to Open Source Projects. Also, many serious OSS projects do have release schedules, testing, etc., often adhered to even more strictly than in many non-OSS projects. Although this model characterizes the nature of an OSS community and OSS project development, the increasing role of OSS in our society requires more. It is necessary to take into consideration that OSS is a form of system development and to find ways that measure quality, success, contribution and also the impact of communication between developers and users.

Since OSS development can be defined as a form of information system development, the study of the success measures of an IS becomes relevant.

Our starting point is the fact of OSS being somehow more like IS than traditional software development. Due to the looser structure of OSS projects, the roles of people and the OSS community are very important. An OSS project can be seen as an organization, that emerges organically bottom up, as opposed to the top down created-by-design structures of traditional organizations where in information systems have been studied. It can also be seen as an open democratic society, where the role of the individuals and the role of the community are very significant in contrast to the pre-established structures in the traditional software development projects. The similarity of OSS projects to IS suggests that we may use the success measures proposed for IS to predict the success of OSS projects.

One of the most cited models of success in IS, shown in Figure 1, is the one developed by DeLone and McLean in [DM92], [CAH03].

The model in Figure 1, provides a schema for clarifying a multitude of measures into six categories, or dimensions, suggesting a model of temporal and causal interdependencies between these categories. DeLone and McLean conclude their paper with the comment that the model in Figure 1 'clearly needs further development and validation before it could serve as a basis for the selection of appropriate IS success measures'.

The model can be interpreted as follows:

Systems quality and *information quality* singularly and jointly affect both *use* and *user satisfaction*. Additionally, the amount of *use* can affect the degree of *user satisfaction* positively or negatively – as well as the other way around. *Use* and *user satisfaction* are direct antecedents of *individual impact*; and lastly this impact on individual performance will eventually have some *organizational impact* (DeLone and McLean, 1992: 83-87).

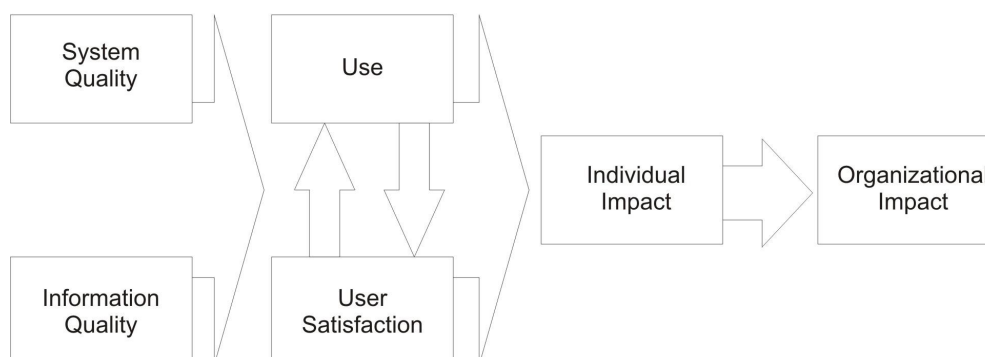


Figure 1: DeLone and McLean's Model of IS Success

But IS is any combination of information technology and people's activities. In traditional software projects, the structures and goals, budgets, schedules, etc., are set more strictly before hand. Thus, aside from the technical competence of the people involved, the "people" involved do not matter that much. The structures do not leave much room for their individuality to change the process or the outcome. In OSS projects, all structures are much looser. Therefore, the characteristics of the individuals, their personal motivations, personalities, goals, etc., play a much more important role in the development process. Actually, we believe that the emphasis of IS on people and their interactions with the software is their contact point with OSS projects. OSS is free, is open source, is built by volunteers and it allows heterogeneous teams. On the other hand traditional software must be paid, its source is closed, is built by the company employees and has a stable development team. Also, and as in IS, an OSS community needs to coordinate development: for example, there is the need for management to know whether each volunteer knows what to do and all this requires decision making.

Considering an OSS project as an IS with particular characteristics, we believe that the DeLone and McLean model fits our purpose to predict the success of OSS projects.

3 Measures for Quality and Success in OSS

Abandoned projects may find an explanation in the volunteers that find more interesting projects but also the successful projects face important problems related to quality due to the not always reliable contribution of volunteers [Mic04]. Also, it must be taken into consideration that OSS communities seek newly and this fact also reflects on the web hosting services. Note that it is possible to find abandoned projects on SourceForge that are active on GitHub or other web hosting.

Many quality problems have been identified in interviews performed in free software communities [VMT07]. Analysis of web hosting for OSS, such as GitHub or SourceForge also provide useful insight. SourceForge is one of the most popular hosting sites for free software with over 313,000 projects. It is not only a good resource to find well maintained projects, but also to find out a large number of abandoned projects and low quality code.

The most common problems identified are the *unsupported code*, *configuration management*,

security updates and *users* that do not know how to report bugs.

- *unsupported code* affects quality. Note that it is impossible to know how to handle code that has previously been contributed but it is now unmaintained;
- *configuration management* affects quality depending on the number of contributions. It is difficult for the lead developer to test all contributions;
- *security updates* affects quality since, although some bugs are fixed there isn't any schedule to make the fixes available;
- *users* who sometimes do not know how to report a bug and often fail to include enough information in bug reports;

All these problems need to be faced and it is necessary to come up with a range of measures that can attest the quality of an OSS product.

Some say that the quality of OSS can be measured by the number of users and the code quality. Actually, in the book "The Cathedral and the Bazaar" [Ray99], Raymond suggests that the level of quality is partly due to the high degree of peer review and user involvement often found in free software projects.

Code quality has been studied extensively in software engineering. Literature provides many possible measures of the quality of software. This includes understandability, completeness, conciseness, portability, maintainability, testability, usability, reliability, structuredness and efficiency, etc (Boehm et al. 1976; Gorton and Liu 2002). Code quality measures seem particularly applicable on OSS, since source code is available by definition. Some authors have already studied this dimension, Stamelos et al (2002) believe that OSS is, in general, of good quality. Mishra (2002) offers an analytic model that suggests factors contributing to OSS quality.

Based only on the literature review, we present what we consider the most relevant measures for OSS success[CAH03]: *User, Product, Development Process, Developers, Use, Recognition and Influence*.

- *user* who have the possibility to change the software to satisfy her needs. Moreover, some users feel involved in the development process through mailing lists or bug reports or other forms of communication with the development team. However, sometimes inconsistency is observed in open source software mainly due to "short feedback loops between users and core developers" [RCA11], [PYM+06]. Users feel that frequent beta versions of the software frustrates some of their expectations. However, the satisfaction of users / clients increases while they feel involved in the development of new releases.
- *product*, whose key success issues are the degree of meeting requirements of design, the code structure organized, clear and maintainable documentation. These aspects are also related to the code quality. A final product with quality has a higher probability to achieve success. Also code quality may be measured by its portability and compatibility with other systems and programs, as mentioned previously. Moreover, if an OSS product is available from more than one source, then the probability of reaching a bigger number of users increase;



- *process* which is complex and requires a high level of activity in the form of fixing bugs, writing, updates, etc. Developers must always work having in mind the goals of the project. Also, after a beta release, it is important to keep track of all bugs and fix them in a reasonable time. Moreover, the level of activity of a product is also a measure of the time the group has been active;
- *developers*, that are the main contributors of OSS. The OSS community of developers has the particular aspect of sharing their developers among several projects. This give each project more opportunities to succeed due to the different background and different expertise behind the contributions; just like users, developers also need to feel satisfaction for innovating. This also gives them the opportunity to enjoy the process of development;
- *use*. Ultimately, success can be measured by the effective use of a certain OSS product. The usage can be measured by the number of users, besides the developers, that use an OSS product and the number of downloads it gets;
- *recognition*, by the peers is also a way to measure success. If other projects or organizations pay attention and recognize the importance of a product then it is more likely that they will refer to it at some point;
- *influence*. Since OSS code is free, the degree of code re-use in other projects can be considered as a measure of success.

4 Communication and Contribution in OSS

It is clear that the OSS phenomena is mainly driven by the developers communities. Although OSS projects differ amongst themselves, for example, in their size, also the communities differ in levels participation in coding and communication. Actually, communication and contribution also depend on having suitable strategies. These strategies have a number of dependencies that can be measured.

We believe that communication and contribution play an important role in OSS development.

Researches who have studied the OSS phenomena and OSS projects have largely focused their attentions on contributions of code and the overall contribution itself. However, in addition to contribution, we believe that also communication plays a relevant role during OSS development. Communication in OSS also contributes to knowledge, sharing and collective innovation.

As mentioned on Section 2, OSS proponents have argued that the strengths of the OSS phenomenon are associated with the bazaar model of full participation from large numbers of developers and users [Ray99]. This argument is based on the rationale that a highly participative community may lead to richer discussions, better flow of ideas, efficient code development, faster bug finding and fixing and, hence, faster and efficient project growth [A.C98], [Ray99], [VMT07]. However, empirical studies have found that the numbers of contributors / developers by themselves are not a sufficient factor to sustain the bazaar view of OSS projects. A.Cox [A.C98], a senior developer of Linux, argues that “the bazaar model may turn into a “town council” where equally participative community produces only noise and hence no output” damaging

communication. He further claims that “such a bazaar - town council model may slow down the evolution of a project...”.

These different points of view raise the importance of communication as well as contribution in OSS development. The impact of non- existing communication and chaotic contribution may destroy a functional democratic OSS community and create an anarchic one compromising OSS success and quality. Existing literature suggests measures for communication and contribution. As far as communication is concerned *stable releases*, *participation*, *interactivity* and *release strategy* are considered measures.

- *Software stable releases* are considered a communication measure even if some authors hold that an OSS project never leaves the Beta state. Stable releases may be considered a measure of communication since new releases typically bring in new users, i.e., the number of downloads should increase compared to previous releases. Also, the release notes improve communication between developers and users, which helps to boost the testing function shared by users;
- *Participation* in developing, testing and use is also an obvious measure;
- *Communication interactivity* is expected to have a positive impact on OSS development. It is believed that a good, interactive communication between developers leads to successful and quality products;
- *Release strategy*, under the typical OSS slogan “release often and release fast”. The advantage of using this mechanism is to provide the necessary feedback to understand if the approach followed in a project is the best.

For contribution, the existing literature suggests the following measures:

- *Development time*, i.e., the length of time between the inception date of the project and the release date of the identified stable release;
- *Code contribution*, taking into account that there are different ways to contribute to an OSS project, and that this should somehow be recorded;
- *Number of developers*, although there is no agreement if the number of developers is a measure of real contribution and successful communication in an OSS project, we will consider it as a measure;
- *Project size*, which may determine the effectiveness of an OSS project. The smaller the project the easier it is to control the constant changes that occur during its development.

5 Quality, Success, Communication and Contribution on OSS

We began by using the information system model of DeLone and McLean[DM92]. Through this model, we present the relationship between quality and success. However, this model is not clear enough to further explore the effects of contribution and communication and their impact on quality and success.

Although some authors, as in “The cathedral and The Bazaar” emphasize the importance of the number of contributors, one cannot analyze only the numbers of users and developers to say that an OSS project has quality and is successful. There is a lack of empirical data to sustain this statement. On the other hand, appearance of these statements in the literature suggests a strong relationship between quality and success in OSS projects.

But, how to link Quality, Success, Contribution and Communication? Do we need another model or is it enough to extend an existing one?

It turns out that, having measures for the four concepts defined, makes it possible to determine a relationship between quality, contribution, communication and success by extending the DeLone and McLean model to include contributors and communication. In section 2, quality and success were presented together because many believe that quality by itself entails a successful product. This is true but also the communication and contribution aspects must be considered. An OSS can be successful, have relatively good quality but, if it lacks the communication part then it will not be as successful as wished. Actually, success is not only measured by the number of contributors that take part in the development team. It is necessary for the developers to be motivated and develop OSS collaboratively.

Also in section 3, communication and contribution were presented together since, it is our belief that it is impossible to have quality contribution without communication between contributors, and in the limit, users.

Communication during OSS development has a fundamental importance and so, we believe the model proposed by DeLone and McLean[DM92] can be extended to deal explicitly with communication and contribution:

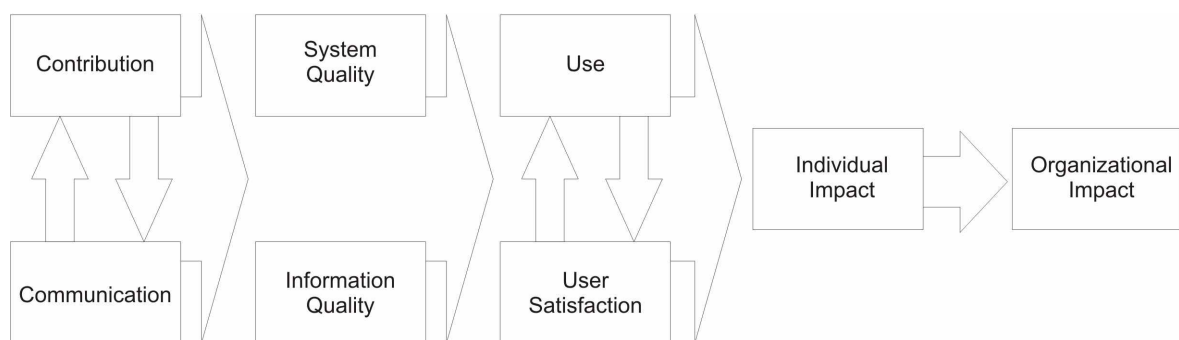


Figure 2: Extended Information System Model

Figure 2 presents an extended version of the DeLone and McLean model which highlights the relevance of communication and contribution together for OSS development. Figure 2 aims to clarify the importance of communication to the contributors in order to develop quality software and achieve success. The 8 dimensions in Figure 2 are interrelated, as it happens in the DeLone and McLean model, wherein the 6 categories of success are proposed as interrelated rather than independent dimensions.

6 Conclusion and Future Work

Open source software projects demonstrate a number of similarities with traditional information systems. However some of their characteristics significantly distinguish them from typical information systems as traditionally understood. The similarities between OSS projects and information systems motivate researching the possibilities of extending IS theories to OSS. This paper makes one such attempt to apply and extend the IS success model of DeLone and McLean to evaluate the success of OSS projects. We presented the concepts and measures for quality, success, contribution, and the characteristics of communication, defining a set of measures that can be used in the analysis of quality and success of OSS projects, augmented with measures for communication and contribution. We also presented an extended version of the DeLone and McLean model in order to make clear the relevance of contribution and communication to achieve quality and success in OSS projects.

DeLone and McLean conclude their paper commenting on the model in Figure 1 as follows: “clearly it needs further development and validation before it could serve as a basis for the selection of appropriate Information System success measures”. Our extended model for OSS projects clearly needs further development and validation before it can serve as a basis for the selection of the appropriate parameters in OSS projects. Nevertheless, we believe that the prospect of having a model to predict the success and impact of OSS projects has interesting practical implications. For instance, one can use such a model to assess the potential success of an ongoing OSS project, and subsequently adjust the structure, organization, participants, contributions, etc., that affect the parameters of the model deliberately, in order to increase the success measure of the project, before it is too late. As future work, we intend to further develop and validate this model.

Bibliography

- [A.C98] A.Cox. Cathedrals, Bazaars and the Town Council. 1998. <http://slashdot.org/>.
- [CAH03] K. Crowston, H. Annabi, J. Howison. Defining Open Source Software Project Success. In *in Proceedings of the 24th International Conference on Information Systems (ICIS 2003*. Pp. 327–340. 2003.
- [DM92] W. Delone, E. McLean. Information Systems Success: The Quest for the Dependent Variable. *Information Systems Research* 3(1):60–95, 1992.
- [Mic04] M. Michlmayr. Managing Volunteer Activity in Free Software Projects. In *Proceedings of the 2004 USENIX Annual Technical Conference, FREENIX Track*. Pp. 93–102. Boston, USA, 2004.
- [Per04] Perry Donham. Ten Rules for Evaluating Open Source Software. 2004. <http://collaborative.ws>.
- [PYM⁺06] A. Porter, C. Yilmaz, A. M. Memon, A. S. Krishna, D. C. Schmidt, A. Gokhale. Techniques and processes for improving the quality and performance of open-source software. *Software Process: Improvement and Practice* 11(6):163–176, May 2006.



- [Ray99] E. S. Raymond. *The Cathedral and the Bazaar*. O'Reilly & Associates, Inc., Sebastopol, CA, USA, 1st edition, 1999.
- [RCA11] A. Raza, L. Capretz, F. Ahmed. Users' perception of open source usability: an empirical study. *Engineering with Computers*, pp. 1–13, May 2011.
[doi:10.1007/s00366-011-0222-1](https://doi.org/10.1007/s00366-011-0222-1)
<http://dx.doi.org/10.1007/s00366-011-0222-1>
- [UPO11] UPOSU. User's perception of open source usability: an empirical study. 2011. <http://eng.uwo.ca>.
- [VMT07] P. Vir, S. Ming, F. Y. Tan. An Empirical Investigation of Code Contribution, Communication Participation, and Release Strategy in Open Source Software Development: A Conditional Hazard Model Approach. 2007.